CanShareFolderW

Be careful of malicious folder substitution

Sean Barnum, Cigital, Inc. [vita[1]]

2007-02-28

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 7138 bytes

Attack Category	• Path spoofing or confusion problem
Vulnerability Categories	• Indeterminate File/Path • TOCTOU - Time of Check, Time of Use
Software Context	
Location	• ntshrui.dll (available only on Windows XP)
Description	Used to determine whether to show the "Share this folder" option in web view. CanShareFolderW is vulnerable to TOCTOU attacks.
APIs	**FunctionName** / **Comments** CanShareFolderW / check
Method of Attack	The key issue with respect to TOCTOU vulnerabilities is that programs make assumptions about atomicity of actions. It is assumed that checking the state or identity of a targeted resource followed by an action on that resource is all one action. In reality, there is a period of time between the check and the use that allows either an attacker to intentionally or another interleaved process or thread to unintentionally change the state of the targeted resource and yield unexpected and undesired results. In the case of CanShareFolderW, an attacker may be able to substitute a shareable folder with a non-shareable one, thereby getting access to folders that should not be available.
Exception Criteria	
Solutions	**Solution Applicability** / **Solution Description** / **Solution Efficacy** Appears to be generally applicable. / Utilize a file descriptor version of check and use / Appears to be generally effective.

1. http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html (Barnum, Sean)

	functions, if possible.	
Generally applicable.	The most basic advice for TOCTOU vulnerabilities is to not perform a check before the use. This does not resolve the underlying issue of the execution of a function on a resource whose state and identity cannot be assured, but it does help to limit the false sense of security given by the check.	Does not resolve the underlying vulnerability but limits the false sense of security given by the check.
Generally applicable.	Limit the interleaving of operations on files from multiple processes.	Does not eliminate the underlying vulnerability but can help make it more difficult to exploit.
Generally applicable.	Limit the spread of time (cycles) between the check and use of a resource.	Does not eliminate the underlying vulnerability but can help make it more difficult to exploit.
Generally applicable.	Recheck the resource after the use call to verify that the action was taken appropriately.	Effective in some cases.

Signature Details	STDAPI CanShareFolderW(LPCWSTR pszPath);

Examples of Incorrect Code	`STDAPI return_value;`
	`LPCWSTR pszpath =`
	`"pszpath_to_folder_to_be_checked"`
	`int check_status;`
	`struct stat statbuf;`
	`[...]`
	`check_status=stat(pszpath,`
	`&statbuf);`
	`[...]`
	`/*Attacker makes change during`
	`this time*/`
	`return_value =`
	`CanShareFolderW(pszpath);`
	`/* Check return value S_OK,`
	`S_FALSE, or HRESULT error */`

Examples of Corrected Code	`STDAPI return_value;`
	`LPCWSTR pszpath =`
	`"pszpath_to_folder_to_be_checked"`
	`[...]`
	`return_value =`
	`CanShareFolderW(pszpath);`
	`/* Check return value S_OK,`
	`S_FALSE, or HRESULT error */`

Source References	• MSDN CanShareFolderW Function[2]
Recommended Resources	**Resource Name** / **Resource Link**
Discriminant Set	**Operating Systems**: • Windows XP Home • Windows XP Pro **Language**

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

1. mailto:copyright@cigital.com